Spring Core\_Maven

**Exercise 2: Implementing Dependency Injection**

**Scenario:**

In the library management application, you need to manage the dependencies between the BookService and BookRepository classes using Spring's IoC and DI.

**Code:**

**pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/maven-v4\_0\_0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.library</groupId>

<artifactId>LibraryManagement</artifactId>

<version>1.0-SNAPSHOT</version>

<packaging>jar</packaging>

<name>LibraryManagement</name>

<url>http://maven.apache.org</url>

<properties>

<maven.compiler.source>1.8</maven.compiler.source>

<maven.compiler.target>1.8</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.36</version>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>exec-maven-plugin</artifactId>

<version>3.1.0</version>

<configuration>

<mainClass>com.library.App2Main</mainClass>

</configuration>

</plugin>

</plugins>

</build>

</project>

**BookRepository2.java**

package com.library.repository;

public class BookRepository2 {

public void save(String bookName) {

System.out.println("[BookRepository2] Saving book: " + bookName);

}

}

**BookService2.java**

package com.library.service;

import com.library.repository.BookRepository2;

public class BookService2 {

private BookRepository2 bookRepository;

// Setter for Dependency Injection

public void setBookRepository(BookRepository2 bookRepository) {

this.bookRepository = bookRepository;

}

public void registerBook(String bookName) {

System.out.println("[BookService2] Registering book: " + bookName);

bookRepository.save(bookName);

}

}

**App2Main.java**

package com.library;

import com.library.service.BookService2;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App2Main {

public static void main(String[] args) {

ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext2.xml");

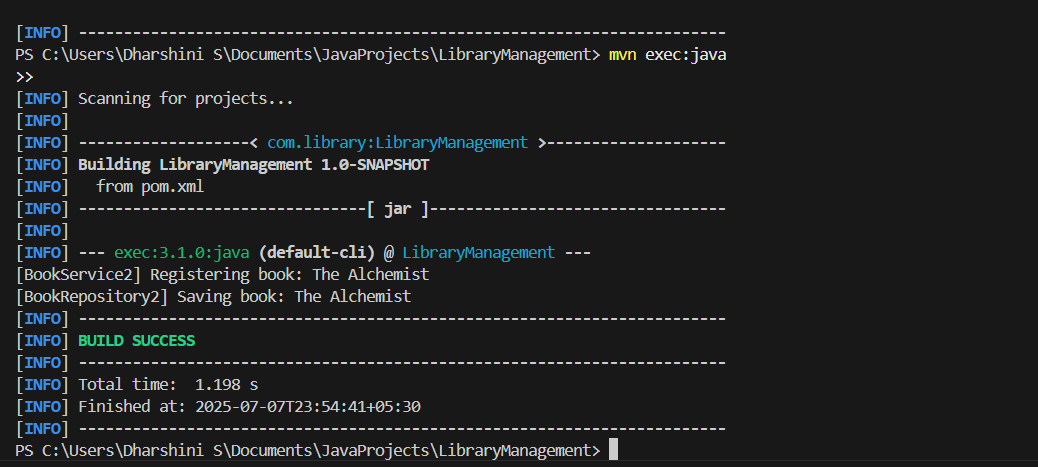
BookService2 service = (BookService2) context.getBean("bookService2");

service.registerBook("The Alchemist");

}

}

**Output:**

****